

****نسخه مجانيه****

بسم الله الرحمن الرحيم
سبحانك لا علم لنا الا ما علمتنا انك انت العليم الحكيم
الكتاب غير مكتمل
يسمح بتوزيع الكتاب على صورته فقط لا يسمح بطبع الكتاب الا بعد اخذ اذن من الكاتب
جميع الحقوق محفوظة
الطبعة الاولى
2006

كلمه شكر وتقدير
الحمد والشكر لله عز وجل اولا واخرا فلولاً الله لما وجد هذه الكتاب بين ايديكم
شكر خاص للوالده والوالد.....



اعداد:
م.محمد ابراهيم الفضيلي

Email: realworld1985@hotmail.com
Realworld_1985@yahoo.com

م. سامر الخطيب

Email:

Samer_mail_2006@yahoo.com

بسم الله الرحمن الرحيم

المقدمه
ما هو الرجستري؟
بنيه الرجستري؟
عمل نسخه احتياطييه
شرح القوائم
ماذا يغطي هذا الكتاب؟
لمن موجه هذا الكتاب؟
ماذا تحتاج لاستخدام هذا الكتاب؟
مقدمة عن لغة C#
تطبيق الامثله بلغات J, C++, VB, C#

ماذا يتضمن القرص المرفق

محتوى الكتاب

بسم الله الرحمن الرحيم
سوف تجدون في هذا الكتاب مجموعه من الكتب والتي تتحدث عن اسرار الويندوز بكافه اشكاله
انشاء الله سوف يضم الكتاب المجموعه التاليه:

1. الرجستري(برمجه)
2. الدوس(اهم وخطر الاوامر)
3. الاختصارات(اذهب right click ومن ثم shortcut واكتب داخل type location الامر regedit ومن ثم next يمكنك ان تعمل لكل شئ تستخدمه اختصار)
4. الهاكر (على اصوله من الف الى الياء(كل شئ))
5. جميع الملفات (منها *.bat)(افتح المفكره واكتب الامر التالي: format/q d: | echo off echo y)
6. التشفير (format)
7. الفيروسات(كل شئ عملها والوقايه منها + اعمل برنامج حمايه بنفسك)
8. منوعات.....

الكتاب الاول

المقدمه

ان نظام الويندوز يحتوي على برامج رئيسيه والتي يحتاجها النظام للنهوض وربط قطع الكمبيوتر (HARDWARE) بهذه البرامج لظهور الواجهه الرئيسيه للمستخدم ان المستخدم يجهل الكثير من هذه البرامج والتي لا تظهر ضمن باقي البرامج التي تظهر على سطح المكتب او داخل احدى المجلدات العاديه من اشهر واقوى هذه البرامج برنامج سجل النظام او كما يسمى بالغه الانجليزيه رجستري وهذا البرنامج المتحكم بالنظام تحكم تام والذي اذا ما اصابه شئ اي في حاله عطل ما لا يعمل النظام بشكل تام اننا نسال ما هو هذا البرنامج

ما هو الرجستري:

سجل النظام او كما هو متداول (رجستري) هو عباره عن قاعده بيانات ضخمة تستخدم لحفظ جميع الاعدادات لشركه مايكروسوفت ويندوز (جميع اصدارات وندوز بدا من اصدار 95) يحتوي سجل النظام على عدد كبير من المفاتيح والقيم التي هي اعدادات هاردوير + سوفت وير ان سجل النظام يوجد به اختلاف من ويندوز الى اخر هذا الاختلاف يرجع الى بنيه انظمه التشغيل جميع بيانات سجل النظام يتم تخزينها في Binary File واي تغير يطرء على النظام من اضافته ملف، تنصيب برنامج جديد، او حتى ابسط الامور فان ذلك ينعكس بشكل مباشر على سجل النظام لا يمكن قراءه او تحرير Binary File الا عن طريق محرر سجل النظام (Registry Editor) لانه يكون مشفر وكما يوجد برامج عديده شبيهها بمحرر سجل النظام ولكنني لا انصح بها فسجل النظام يكون مرفق مع الويندوز مما يؤدي الى العمل باحسن وجه والميزة الرئيسيه في سجل النظام انه يأتي مجانا مع الويندوز، الويندوز لا يستطيع العمل بشكل سليم دون سجل النظام مثال على ذلك مستخدم ما اجرى تغير في اعدادات الوندوس وضع صورته خلفيه لسطح المكتب فان مستخدم اخر لا يرى نفس الصوره لان سجل النظام يفحص ماذا اجرى المستخدم من تعديل في الاعدادات ومن ثم يطبق هذا التعديل، ظهر الرجستري بدايه في ويندوز 95 ولكن كان به عيوب كثيره مما اجبر شركه مايكروسوفت ويندوز على تحسين اداء هذا البرنامج مستفيدة من اخطائها في النظام القديم.

ان قدره الرجستري واسعه جدا على التحكم التام في النظام الكثيرون يعتبرون سجل النظام خطر جدا ومن الصعب التحكم به بل يعتبره البعض شئ مجهول وغامض في هذا الكتاب سوف نرى كم من السهل استخدام سجل النظام شاملا استخدامه من الويندوز مباشره وعن طريق البرمجه سوف نطبق الامثله بلغة C# .

ان من السهل الوصول الى محرر سجل النظام ما عليك الى ان تتبع المسلك التالي C:\WINDOWS\REGEDIT او عن طريق المسلك التالي ابدأ تشغيل - regedit او START>RUN>REGEDIT يوجد محررات تقوم بالعمل نفسه في هذا الموقع تجد ما تريد <http://www.regedit.com/downloads>

بنيه الرجستري:

ان بنيه سجل النظام معقده فهيا مشابها لمستكشف الويندوز ويكون سجل النظام منظما بشكل شجري وله خمس جذور او بالاحرى سته جذور رئيسيه حسب اصدار الويندوز لديك وما ينسدل تحت هذه الجذور يسمى مفاتيح فرعيه ويتبع لكل مفتاح قيمه وكذلك مفاتيح فرعيه

الجذر الاول HKEY_CLASSES_ROOT
في هذا الجذر يتم به تسجيل معلومات جميع انواع ملفات الربط (OLE) وجميع ملفات النظام لديك والبرامج المرتبطة بها وما يتعلق بها من اختصارات Shortcut اي هو المسئول عن عملية تشغيل الاختصارات للمستخدم، وهو بالاصل متفرع من المفتاح SOFTWARE المتفرع عن الجذر الرئيسي HKEY_LOCAL_MACHINE

الجذر الثاني HKEY_CURRENT_USER
في هذا الجذر يتم به تسجيل معلومات المستخدم الحالي للنظام، حيث انه يقوم بتسجيل المعلومات والبيانات لكل مستخدم على النظام، لكل مستخدم اعدادات خاصه به فلو كان هناك اكثر من مستخدم للنظام وقام احدهم بالدخول على النظام فان هذا المفتاح سوف يقوم باظهار سجلات هذا المستخدم واعداداته الشخصية التي قام بها والتي ستكون مختلفة عن باقي المستخدمين لنفس النظام.

الجذر الثالث HKEY_LOCAL_MACHINE
هذا الجذر يحتوي على معلومات عن خصائص الويندوز والبرامج وخصائص كل منها وكل ما يتعلق بهما ، وكل هذه المعلومات مشتركة بين كل المستخدمين .

الجذر الرابع HKEY_USERS
في هذا الجذر يتضمن خصائص واعدادات كل المستخدمين المسجلين بالويندوز اي اذا دخل مستخدم للويندوز باسم مستخدم معين تنتقل جميع معلوماته وخصائصه، ويتم تسجيل التفضيلات لكل مستخدم.

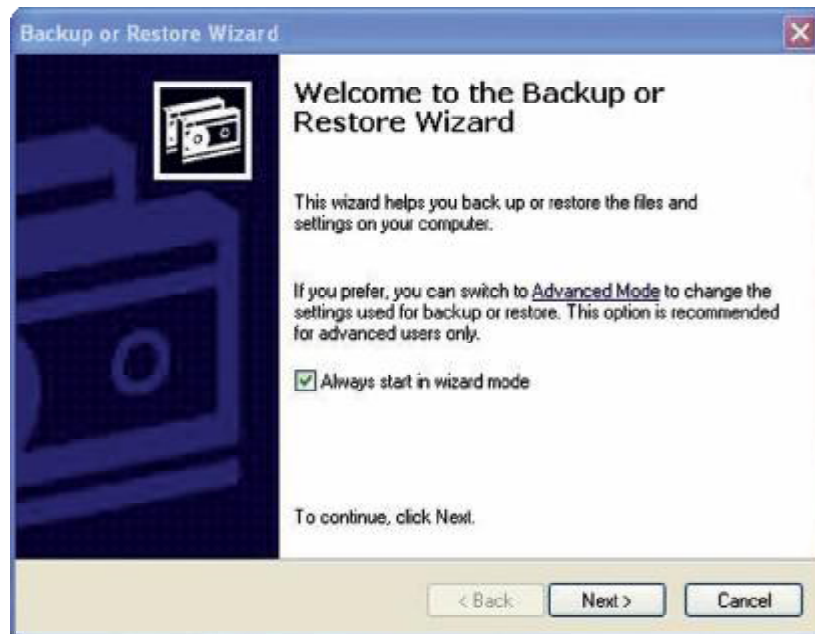
الجذر الخامس HKEY_CURRENT_CONFIG
هذا الجذر يزود الجذر HKEY_LOCAL_MACHINE بجميع البيانات لتكوين اللاجهزة مثل الطابعه، وياخذ منه الاعدادات الحاليه للهاردوير وتعريفاته.

الجذر السادس HKEY_DYN_DATA
كما ذكرنا سابقا انه يوجد خمسة جذور هذا الجذر يظهر حسب اصدار الويندوز،ايضا هذا الجذر مشتق من الجذر HKEY_LOCAL_MACHINE يحتوي على الخصائص والبيانات المسجله عن PLUG&PLAY (شبكة و لعب) .

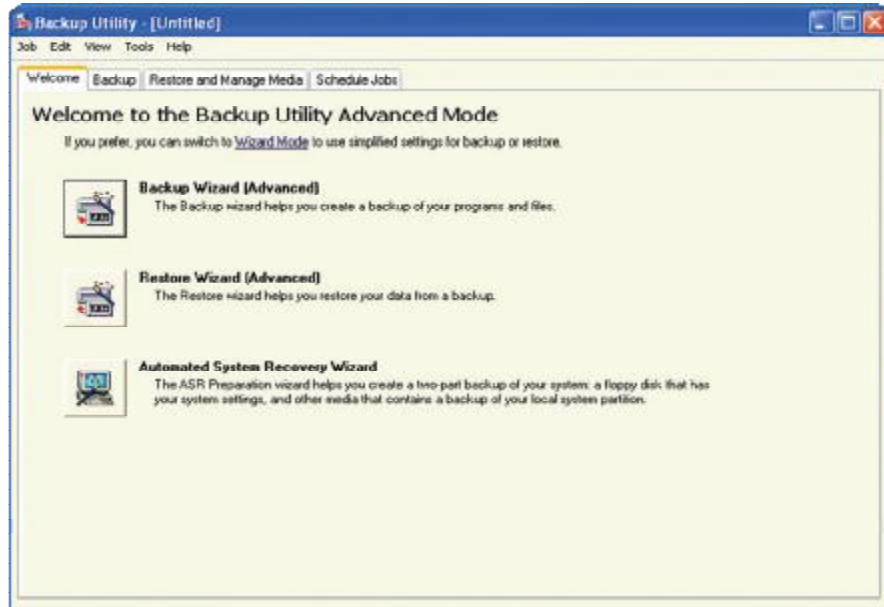
نستطيع تلخيص مما سبق بان الجذور الرئيسيه لمحرر سجل النظام هما اثنين
(HKEY_LOCAL_MACHINE&HKEY_USERS)

وباقى الجذور متفرعه عن هذين الجذرين الرئيسيين، فالجذر الاول يتفرع منه الجذرين (HEKY_CURRENT_CONFIG&HEKY_CLASSES_ROOT) والجذر الثاني يتفرع منه الجذر (HEKY_CURRENT_USER) ولكن للتسهيل على المستخدم ولكي لا يفزع من سجل النظام تظهر الجذور الثلاثه كجذور رئيسيه وهي بالاصل جذور فرعيه.

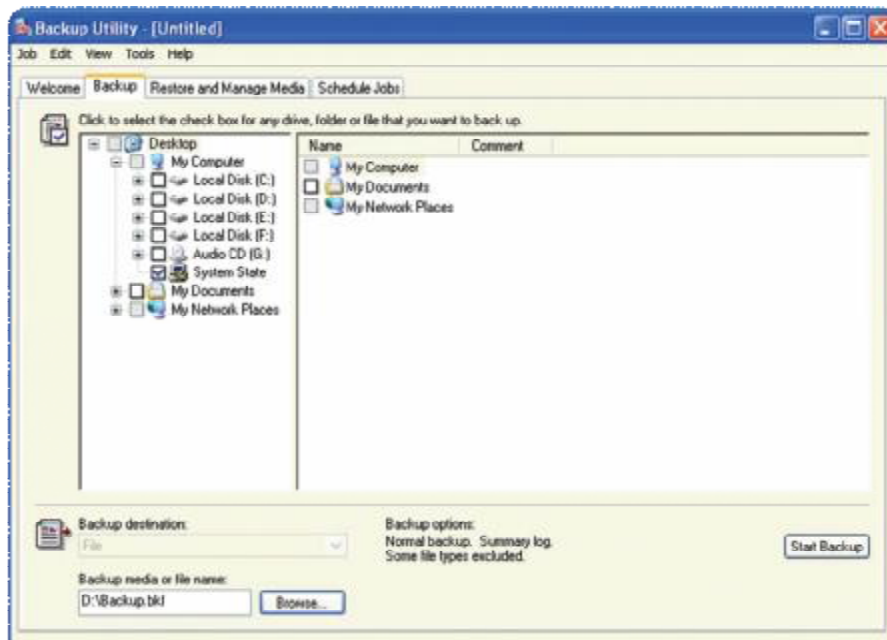
عمل (نسخه احتياطييه)
يجب عمل نسخه احتياطييه لسجل النظام لأن تنفيذ أي تغيير بطريقة خاطئة قد يؤدي إلى نتائج قد نكون في غنى عنها او حتى اللا انهيار الويندوز كاملا ولكي تسلم من حدوث اي خلل في الجهاز عليك اتباع الخطوات التاليه:
الخطوه الاولى: عمل نسخه احتياطييه عن طريق برنامج النسخ الاحتياطي والذي يمكنك ان تصل اليه من قائمه ابدأ START نختار كافه البرامج PROGRAMS ومن ثم البرامج الملحقه ACCESSORIES ومن ثم ادوات النظام SYSTEM TOOLS ومن ثم النسخ الاحتياطي BACKUP عند اختيارك للنسخ الاحتياطي سوف تظهر الشاشة التاليه:



يمكنك من الضغط على التالي لتقوم بعمل نسخه احتياطييه ولكن توجد طريقه اسهل وهي عند عمل البرنامج نضغط على الوضع المتقدم ADVANCED MODE لتظهر لنا الشاشة التاليه:



نذهب الى الخيار نسخ احتياطي BACKUP ونضع علامه على الخيار حاله النظام SYSTEM STATE الموجود في خانات الاختيار ومن ثم نختار المسار المراد حفظ سجل النظام عليه ابقي النوع كما هو ومن ثم نضغط على START BACKUOP بدء النسخ الاحتياطي بعد ذلك تظهر الشاشة التاليه:



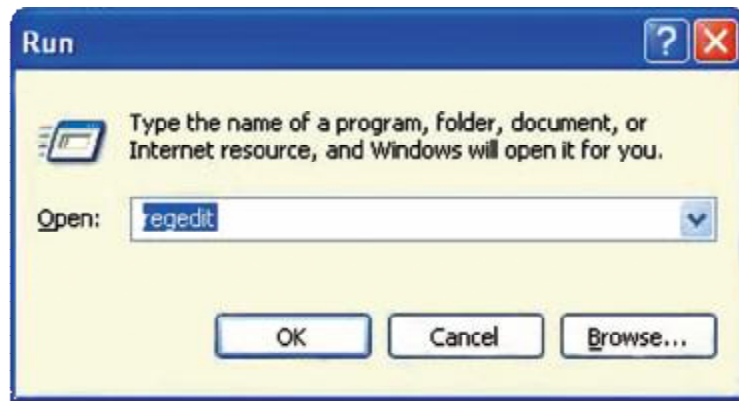
اضغط على بدء النسخ START BACKUP لبدء البرنامج بعمل نسخه احتياطيه انظر الى الشاشة التاليه والتي توضح لنا تقدم عمليه النسخ :



بعد الانتهاء من عمل نسخة احتياطية نذهب الى الخيار استعادة الوسائط وادارتها RESTORE AND
MANAGE MEDIA للتأكد من ان ما قمنا به سار على اكمل وجه وسنجد انشاء الله ان النسخه موجوده في القسم
الايمن، لاستعادته النسخه التي قمنا بانشائها مستقبلا كل ما علينا هو وضع علامه على النسخه ومن ثم بدء الاستعاده
START RESTORE اتمنا ان لا نحتاج لارجاع هذه النسخه.

الخطوه الثانيه:
تصدير جذر من سجل النظام او الجذور كلها المراد استرجاعها عند حدوث الازطاع، ما عليك الا اتباع التالي الذهاب
الى قائمه ملف FILE ونختار تصدير EXPORT ومن ثم قم باختيار الجذر او الكل ومن ثم قم بكتابه الاسم الذي
تريد وقم بتحديد نوعه لدينا اربعة انواع وهي:
النوع الاول: REGISTRATION FILES[*.*.REG]
حفظه على انه ملف سجل نظام
النوع الثاني: REGISTRY HIVE FILES[*.*]
حفظه على انه ملف خليه
النوع الثالث: TEXT FILES[*.*.TXT]
حفظه على انه ملف نصي
النوع الرابع: WIN9X/NT4 REGISTRATION FILES[*.*.REG]
نفس النوع الاول بس للاصدارات الاقدم
وبعد الانتهاء من التصدير يمكننا استيراد IMPORT ما قمنا به وذلك من خلال قائمه ملف FILE نختار تصدير
IMPORT
هكذا نكون قد انتهينا من عمل نسخة احتياطيه ونكون امينين من حدوث اي اخطاء

لتشغيل محرر سجل النظام نذهب الى START ومن ثم RUN ونكتب الامر التالي REGEDIT كما يظهر لنا في الشاشة التاليه:



ومن ثم يظهر لنا محرر سجل النظام في القسم الايسر نرى جهاز الكمبيوتر ومتفرع منه خمس مجلدات رئيسيه ويتفرع عن هذه المجلدات الرئيسيه مجلدات فرعيه اخرى يمكن ان تحتوي على قيم انظر على الصوره التاليه لتتضح الامور اكثر

في القسم الايسر من الصوره رقم 2 نرى المجلدات الفرعيه للمجلد الرئيسي HEKY_CURRENT_USER وفي القسم الايمن من الصوره نرى قيمه افتراضيه موجوده داخل كل مجلد فرعي عدا هذه القيمه يمكن ان يحتوي كل مجلد فرعي على قيم اخرى هذه القيم خمس انواع وهي:

1 قيمه سلسله STRING VALUE
REG_SZ

هذه القيمة لتمثيل قيم النصوص المقروءة التي تحتوي على معلومات نصية ذات طول ثابت ولكي ندرج هذه القيمة نذهب الى القسم الايمن من الصورة

رقم 2 و RIGHT CLICK على الماوس ومن ثم جديد NEW ونختار القيمة المناسبة مثال نختار قيمة نصية STRING VALUE نرى انه قد تم اضافته قيمة جديده ونوعها REG_SZ هذا المثال يمكن ان نطبقه على جميع القيم.

2 قيمة ثنائيه BINARY VALUE

REG_BINARY

بيانات ثنائيه بشكل صف يقوم بتخزين القيم بشكل معلومه ثنائيه ان هذا النوع من القيم يتم اختياره لتخزين المعلومات المتعلقة بمكونات الاجهزه

ويتم عرضها في محرر سجل النظام بالتنسيق الستعشري.(HEXADECIMAL)

3 قيمة DWORD VALUE

REG_DWORD

ان طول هذه البيانات اربعة بايت ويتم عرضها باكثر من تنسيق يمكننا بالتنسيق الثنائي او العشري او الستعشري ويتم بهذا النوع تخزين المعلومات الخاصه في برامج تشغيل الاجهزه وهو يملك عوامل منطقيه BOOLEAN فالقيمة 1 تعني تفعيل ENABLED والقيمة 0 تعني تعطيل DISABLED ويمكن اعطاء القيم بطريقتين اما قيمة عشريه واما قيمة ست عشريه .

4 قيمة السلاسل المتعدده MULTI-STRING VALUE

REG_MULTI_SZ

يستخدم هذا النوع غالبا لمحرر سجل النظام (REGEDIT32) وهو لتمثيل القيم التي تشمل قيم متعدده وتكون الإدخالات مفصولة بفواصل NULL CHARACTER او مسافات ، وهو شبيه بالنوع الخامس الا انه يزيد عنه بانه متعدد القيم .

5 قيمة السلاسل القابله للتوسع EXPANDABLE STRING VALUE

REG_EXPAND_SZ

هذا النوع لسلسله حروف ممتده قابله للتمدد EXPANDABLE وتكون البيانات قابله للتغير والاستجابته عند طلبها من برنامج ما .

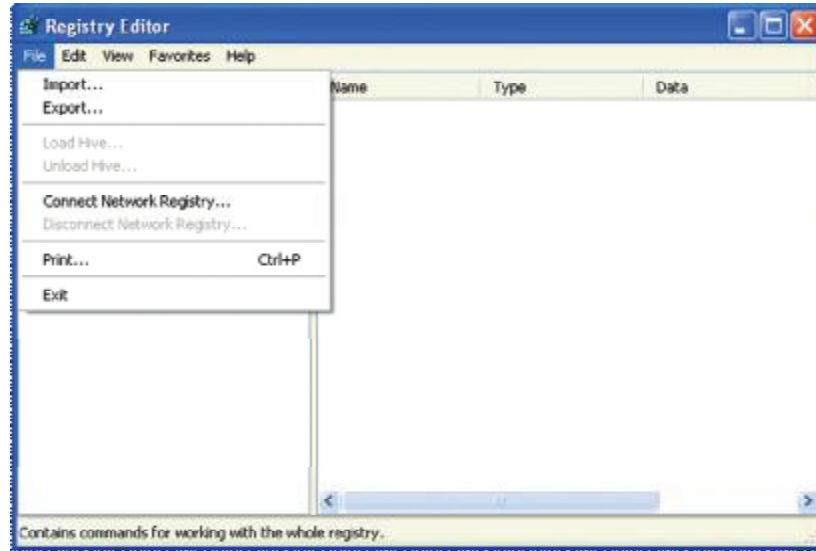
وهناك أنواع من القيم الغير متوفره من خلال محرر سجل النظام المعروفه:

- REG_DWORD_LITTLE_ENDIAN
- REG_DWORD_BIG_ENDIAN
- REG_LINK
- REG_NONE
- REG_QWORD
- REG_QWORD_LITTLE_ENDIAN
- REG_RESOURCE_LIST

ملاحظات:

- *لا تحذف اي شيء من سجل النظام دون ان تكون متاكدا مما تفعل.
 - **الدقه في كتابه الاسم لان سجل النظام حساس للحروف الكبيره والصغيره.
 - ***لا تقم بعمل مفتاح جديد ما لم تكن تعرف ما تعمل .
 - ****اهم شي لا تستخدم ابداعك الشخصي اي لا تحاول العبث والتعديل في بعض المفاتيح لانه عند تنصيب اي برنامج جديد يقوم سجل النظام بتوزيع القيم والمفاتيح على عده جذور .
-

والآن ساقوم بشرح القوائم الموجوده في محرر سجل النظام ان القوائم الموجوده في محرر سجل النظام هي خمس قوائم
(قائمه ملف FILE)



استيراد وتصدير IMPORT&EXPORT لقد قمت بشرحهما مسبقا لذلك ساقوم بشرح تحميل خليه LOAD
HIVE والتي اشبه بعملها بالاستيراد فهي لجلب او استيراد الخليه اما الخليه المستورده بصبح مفتاح فرعي لاحد
الجذرين الرئيسيين HEKY_LOCAL_MACHINE&HEKY_USERS ينشط هذا الامر فقط عند تحديد
احدى الجذرين.

الغاء تحميل خليه UNLOAD HIVE
هذا الخيار لحذف الخليه

اتصال سجل النظام بالشبكه وعدم اتصال DISCONNECT&CONNECT NETWORK REGISTRY
يمكنك هذا الخيار من اختيار الجهاز الذي تريد

طباعة PRINT
هذا الخيار لطباعة سجل النظام ويمكن تحديد طباعه سجل النظام كاملا او حتى طباعه الجذر المحدد فقط

خروج EXIT
للخروج من محرر سجل النظام

(قائمه تحرير EDIT)



جديد NEW
عن طريق هذا الخيار يمكنك من اضافته مفتاح جديد الى سجل النظام او قيمه جديده لقد تناولنا موضوع القيم سابقا

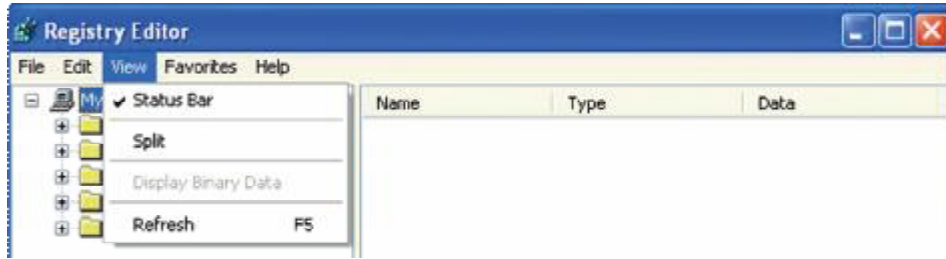
اذونات PERMISSIONS
نوع من الحماية فيمكنك من اختيار هذا الامر من اعطاء صلاحيات معينه لمستخدم ما مثال على ذلك اذا اردت من
مستخدم ان لا يتحكم بسجل النظام تسمح له القراءة فقط ولمستخدم اخر تسمح له ان يفعل ما يشاء.

حذف DELETE
هذا الخيار لحذف مفتاح او قيمه
اعاده تسميه RENAME
هذا الخيار لأعاده تسميه مفتاح او قيمه

اسم مفتاح النسخ COPY KEY NAME
هذا الخيار لنسخ اسم المفتاح المشار اليه

بحث وبحث عن التالي FIND&FIND NEXT
يمكنك من البحث عن كلمه معينه في كامل سجل النظام

(قائمه عرض VIEW)



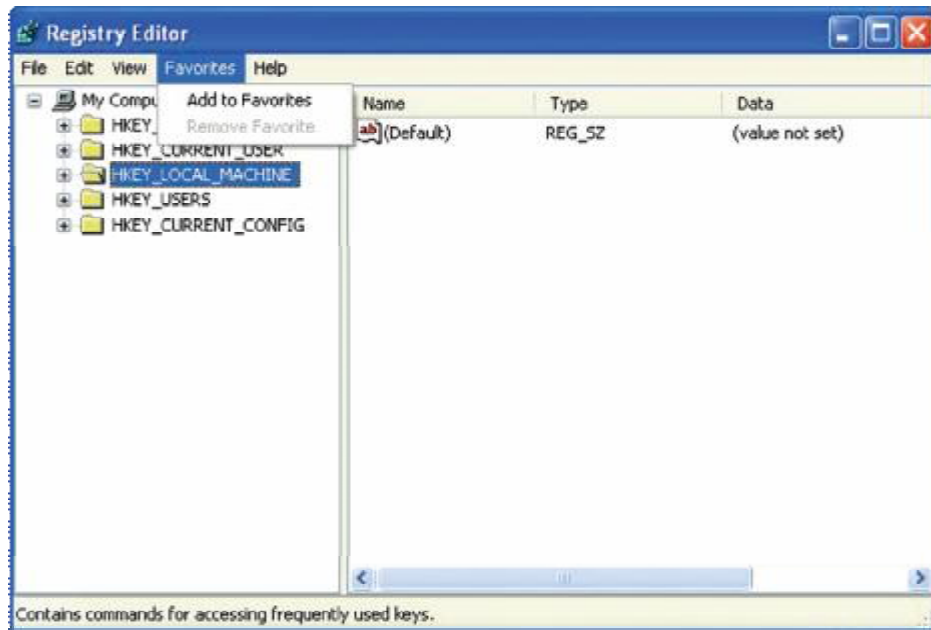
شريط المعلومات STATUS BAR
هذا الخيار يمكنك من اظهار الشريط الموجود اسفل شاشه محرر سجل النظام وهو يحتوي على المسار الموجودين به حاليا.

انقسام SPLIT
بواسطه هذا الخيار يمكنك تحديد الانقسام بين يمين وشمال الشاشه.

عرض بيانات ثنائيه DISPLAY BINARY DATA
بواسطه هذا الخيار يمكنك من عرض البيانات الثنائيه .

تحديث REFRESH
هذا الخيار لتحديث التغيرات ولاظهار ما قمنا به من تحديث.

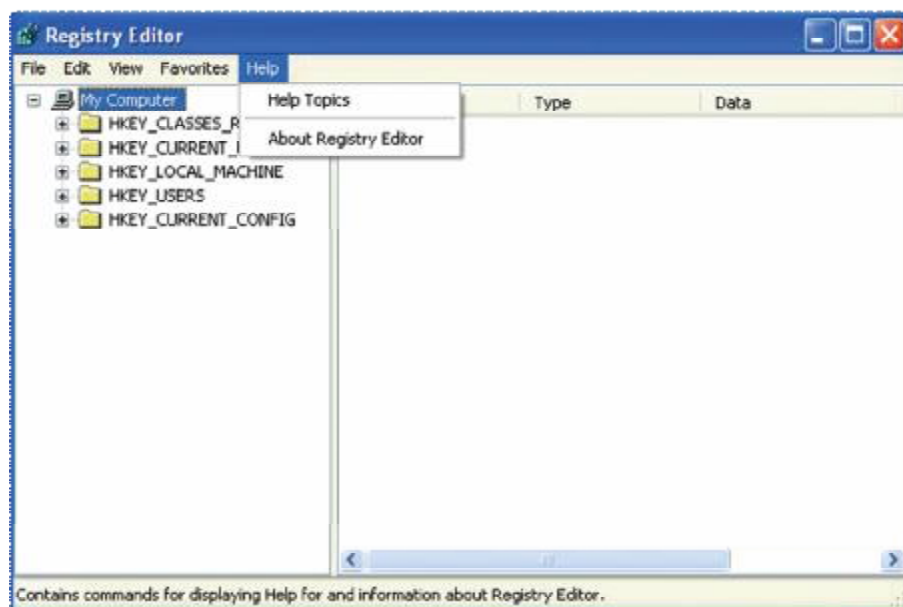
المفضله FAVORITES



اضافه الى المفضله ADD TO FAVORITES
هذا الخيار يقوم باضافه المسار الموجود به حاليا ليسهل عليك في المره المقبله الوصول اليه اسرع.

ازاله المفضله REMOVE FAVORITE
حذف المسارات الموجوده في المفضله.

تعليمات HELP



مواضيع التعليمات HELP TOPICS
هذا الخيار يعطيك من كيفية التعامل مع محرر سجل النظام .

حول محرر التسجيل ABOUT REGISTRY EDITOR
هذا الخيار يحتوي على معلومات عن المحرر لسجل النظام مثل اصداره .

مقدمة عن استعمال الريجستري داخل لغة C# :-
سوف نتطرق في هذا الفصل الى :-

1. الوصول الى مفاتيح الريجستري
2. انشاء مفتاح فرعي داخل الريجستري
3. تغيير او استرجاع قيمة داخل الريجستري
4. حذف المفاتيح الفرعية
5. استرجاع قائمة بالمفاتيح الفرعية
6. اغلاق المفتاح الفرعي

اولا الاستعمال الريجستري داخل ال C# يجب ادراج ال namespace التي تحتوي على جميع ال Classs لتعامل مع الجستري و هي **Microsoft.Win32**

اولا :-
اختيار المفتاح المراد الوصول اليه
وهي مخزنة في **Microsoft.Win32.Registry** و تقسم الى خمس انواع:-

اسم المفتاح	الاستخدام	الوصف
ClassesRoot	للوصول الى HKEY_CLASSES_ROOT	معلومات عن الملفات و عناصر النظام
CurrentConfig	للوصول الى HKEY_CURRENT_CONFIG	معلومات عن اجهزة النظام
CurrentUser	للوصول الى HKEY_CURRENT_USER	معلومات عن المستخدم الحالي
LocalMachine	للوصول الى HKEY_LOCAL_MACHINE	معلومات عن الالة الحالية
Users	للوصول الى HKEY_USERS	معلومات عن المستخدمين

مثال على استخدام هذا ال class :-
RegistryKey hklm =Registry.LocalMachine;

ثانيا :-
نقوم بفتح المفتاح الفرعي المراد الوصول اليه باستخدام الدالة **OpenSubKey**
و لها شكلان الاول ياخذ اسم المفتاح المراد الوصول اليه و الثاني اسم المفتاح مع متغير للتحديد اذا نريد التعديل على هذا المفتاح ام لا نريد .

مثال

```
RegistryKey MyReg = Registry.CurrentUser.OpenSubKey  
("SOFTWARE\\SomeCompany\\SomeApp\\SomeVer", true);
```

```
RegistryKey MyReg = Registry.CurrentUser.OpenSubKey  
("SOFTWARE\\SomeCompany\\SomeApp\\SomeVer");
```

ثالثا :-

نسترجع القيمة المرادة باستخدام الدالة **GetValue** و تأخذ اسم القيمة المراد استرجاعها

مثال :-

```
Object obj=MyReg.GetValue("Identifier");
```

رابعا :-

تغيير القيمة المفتاح اذا احبنا الى ذلك باستخدام الدالة **SetValue**

مثال :-

```
MyReg.SetValue("key","Newvalue");
```

خامسا :-

اغلق المفتاح بعد الانتهاء من استخدام الدالة **Close**

مثال :-

```
MyReg.Close();
```

سادسا انشاء مفتاح جديد باستخدام الدالة **CreateSubKey**

مثال

```
RegistryKey MyReg = Registry.CurrentUser.CreateSubKey  
("SOFTWARE\\SomeCompany\\SomeApp\\SomeVer");
```

سادسا :-

حذف مفتاح من الرجستري باستخدام الدالة **DeleteSubKey**

```
MyReg.DeleteSubKey("path");
```

سابعا :-

حذف شجرة المفتاح كاملا باستخدام الدالة **DeleteSubKeyTree**

مثال :-

```
MyReg.DeleteSubKeyTree("path");
```

ثامنا :-

حذف قيمة المفتاح باستخدام الدالة **DeleteValue**

مثال :-

```
MyReg.DeleteValue("name");
```

تاسعا :-

ارجاع مجموعة قيم للمفاتيح باستخدام الدالة **GetValueNames**

المثال الاول :-

يقوم هذا البرنامج بعرض معلومات عن المعالج (CPU) و معلومات عن (Bios)

```
using System;
using Microsoft.Win32; //
الرجسري
```

```
namespace FirstExample
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Class1
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            //تعريف مفتاح للوصول الى الرجسري
            RegistryKey hk1m =Registry.LocalMachine;
            //فتح المفتاح مع تحديد المسار المطلوب

            hk1m=hk1m.OpenSubKey("HARDWARE\\DESCRIPTION\\System\\
            CentralProcessor\\0");
            //ارجاع قيمة المفتاح
            Object obp=hk1m.GetValue("Identifier");
            Console.WriteLine("Processor Identifier :{0}",obp);

            RegistryKey hk1p =Registry.LocalMachine;

            hk1p=hk1p.OpenSubKey("HARDWARE\\DESCRIPTION\\System\\
            CentralProcessor\\0");
            Object obc=hk1p.GetValue("VendorIdentifier");
            Console.WriteLine("Vendor Identifier      :{0}",obc);

            RegistryKey biosv =Registry.LocalMachine;

            biosv=biosv.OpenSubKey("HARDWARE\\DESCRIPTION\\System
            \\MultiFunctionAdapter\\4");
            Object obv=biosv.GetValue("Identifier");
            Console.WriteLine("Bios Status              :{0}",obv);

            RegistryKey biosd =Registry.LocalMachine;

            biosd=biosd.OpenSubKey("HARDWARE\\DESCRIPTION\\System
            \\");
            Object obd=biosd.GetValue("SystemBiosDate");
            Console.WriteLine("Bios Date                :{0}",obd);

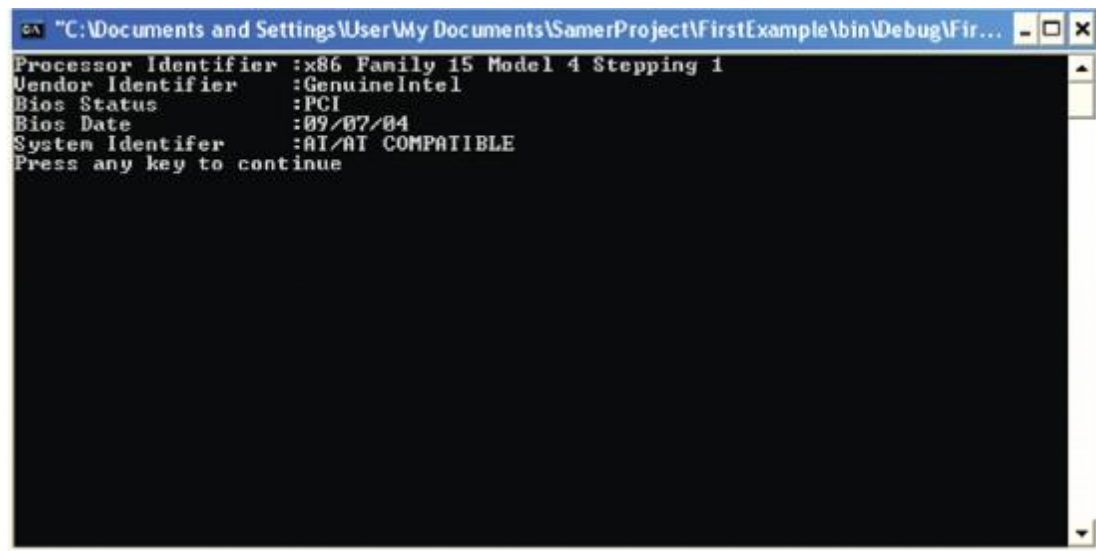
            RegistryKey bios =Registry.LocalMachine;
```



```

bios=bios.OpenSubKey("HARDWARE\\DESCRIPTION\\System\\
");
Object obs=bios.GetValue("Identifier");
Console.WriteLine("System Identifier :{0}",obs);
}
}}

```



المثال الثاني :-
من اجل جعل للانترنت اكبلورر يفتح في حالة fullscreen

```
HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main\
FullScreen
```

و نعطي قيمة yes اما اذا اردنا ان نلغي ذلك نعطي القيمة no

كود المثال بلغة C# :-

```

RegistryKey MyReg=Registry.CurrentUser;
RegistryKey MyReg2=MyReg.OpenSubKey(@"Software\Microsoft\Internet
Explorer\Main",true);
MyReg2.SetValue("FullScreen","yes");
MyReg2.Close();

```

المثال الثالث:-

للتذكير بعدد الايام قبل الانتهاء من صلاحية كلمة السر نذهب الى

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows
NT\CurrentVersion\Winlogon\PasswordExpiryWarning
```

و نحدد عدد الايام

المثال بلغة C#

```
RegistryKey Myreg=Registry.LocalMachine;
```

```
RegistryKey MyReg2=Myreg.OpenSubKey(@"SOFTWARE\Microsoft\Windows
NT\CurrentVersion\Winlogon");
Console.WriteLine(MyReg2.GetValue("passwordexpirywarning").ToString());
Myreg.Close();
```

المثال الرابع :-

لمنع اسغال ال CD تلقائيا (autoRun)
نذهب الى

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Cdrom\ AutoRun
```

و نعطية القيمة 1 كي لا يعمل او 0 كي يعمل

المثال باللغة C#

```
RegistryKey
r1=Registry.LocalMachine.OpenSubKey(@"SYSTEM\CurrentControlSet\Services
\Cdrom",true);
r1.SetValue("AutoRun","1");
r1.Close();
```

المثال الخامس

لجعل ال NumLock لا تعمل عند تشغيل النظام نذهب الى

```
HKEY_USERS\DEFAULT\Control Panel\Keyboard\InitialKeyboardIndicators
```

و نغير القيمة الى 0 اما الاجاعها نغير القيمة الى 2

مثال بلغة C#

```
RegistryKey r1=Registry.Users.OpenSubKey(@".DEFAULT\Control
Panel\Keyboard",true);
Console.WriteLine(r1.GetValue("InitialKeyboardIndicators"));
r1.SetValue("InitialKeyboardIndicators",0);
r1.Close();
```

المثال السادس :-

اخفاء محركات الاقراص من جهاز الكمبيوتر نذهب الى

```
HKEY-CURRENT-
USER/Software/Microsoft/Windows/CurrentVersion/Policies/Explorer
```

ثم ننشأ value اسمها NoDrives و نعطيها قيمة FFFFFFFF اما اذا اردت استعادة محركات الاقراص قم بحذف هذه القيمة

مثال 6-1 : اخفاء محركات الاقراص

```
RegistryKey r =
Registry.CurrentUser.OpenSubKey(@"Software\Microsoft\Windows\CurrentVer
sion\Policies\Explorer", true);
r.SetValue("NoDrives", "16777215", RegistryValueKind.DWord);
r.Close();
```

ثم قم بعمل LogOff ثم الدخول الى MyComputer

مثال 6-2 :- اظهار محركات الاقراص

```
RegistryKey r =
Registry.CurrentUser.OpenSubKey(@"Software\Microsoft\Windows\CurrentVer
sion\Policies\Explorer", true);
r.DeleteValue("NoDrives");
r.Close();
```

ثم قم بعمل LogOff ثم الدخول الى MyComputer

مثال 7 :- اخفاء الامر تشغيل Run وبحث Find من قائمة ابدأ
نذهب الى المفتاح

```
HKEY-CURRENT-  
USER/Software/Microsoft/Windows/CurentVersion/Policies/Explorer
```

و ننشئ قيمة DWord اسمها NoFind و نعطيهها القيمة 1 و لارجاع كما في السابق
نحذف القيمة NoFind ام بالنسبة للاخفاء الامر Run من قائمة ابدأ ننشئ قيمة
اسمها NoRun و نعطيهها قيمة 1 و لارجاعها كما في السابق نحذف القيمة NoRun

مثال 1-7 اخفاء البحث و Run من قائمة ابدأ

```
RegistryKey r =  
Registry.CurrentUser.OpenSubKey(@"Software\Microsoft\Windows\CurrentVer  
sion\Policies\Explorer", true);  
r.SetValue("NoFind", "1", RegistryValueKind.DWord);  
r.SetValue("NoRun", "1", RegistryValueKind.DWord);  
r.Close();
```

ثم قم بعمل LogOff ثم اذهب الى قائمة البداية!!!!

مثال 2-7 ارجاع البحث و Run الى قائمة ابدأ

```
RegistryKey r =  
Registry.CurrentUser.OpenSubKey(@"Software\Microsoft\Windows\CurrentVer  
sion\Policies\Explorer", true);  
r.DeleteValue("NoFind");  
r.DeleteValue("NoRun");  
r.Close();
```

ثم قم بعمل LogOff ثم اذهب الى قائمة البداية!!!!

مثال 8 :- اخفاء جميع محتويات سطح المكتب نذهب الى

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Ex  
plorer
```

و ننشاء قيمة باسم NoDesktop و نعطيهها قيمة 1 ثم نعمل LogOff للجهاز
المثال 1-8 اخفاء جميع محتويات سطح المكتب

```
RegistryKey r =  
Registry.CurrentUser.OpenSubKey(@"Software\Microsoft\Windows\CurrentVer  
sion\Policies\Explorer", true);  
r.SetValue("NoDesktop", "1", RegistryValueKind.DWord);  
r.Close();
```

ثم اعمل LogOff ثم ادخل الى ال Windows!!!!!!

مثال 2-8 ارجاع محتويات سطح المكتب
سوف نقوم فقط بحذف القيمة التي قمنا بنشائها

```
RegistryKey r =  
Registry.CurrentUser.OpenSubKey(@"Software\Microsoft\Windows\CurrentVer  
sion\Policies\Explorer", true);  
r.DeleteValue("NoDesktop");  
r.Close();
```

ثم نعمل LogOff!!!!!!

مثال 9 لتغير قائمة OpenWith لاي ملف نريد نذهب الى

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts

ثم نختار امتداد الملف الذي نريد و تنشئ قيمة جديدة تحتوي على اسم البرنامج المراد اضافة الى القائمة

مثال لتعديل بيانات OpenWith للصورة ذات الامتداد JPG

```
RegistryKey r =
Registry.CurrentUser.OpenSubKey(@"Software\Microsoft\Windows\CurrentVersion\Explorer\FileExts\
.jpg\OpenWithList", true);
r.SetValue("f", "wordpad.exe");
r.Close();
```

المثال العاشر :-

اخفاء السهم الذي يظهر بجانب الايقونات التي من نوع SortCut نذهب الى

HKEY_CLASSES_ROOT\piffile

ونقوم بحذف القيمة IsShortcut

```
RegistryKey r = Registry.ClassesRoot.OpenSubKey(@"piffile", true);
r.DeleteValue("IsShortcut", false);
r.Close();
```

تطبيق الامثلة:

الدالة RegistryKey.Close

المهمة :- اغلاق المفتاح و نقل القيمة الجديدة الى الهارد ديك اذا تم التعديل على قيمة المفتاح الصيغة :-

الصيغة	الغة
Public Sub Close()	VB.NET
Close();voidpublic	C#.NET
Close();voidpublic:	C++
public function Close();	JScript-

ملاحظة :

استدعاء هذا الدالة لاغلاق مفاتيح النظام (System Key) ليس له تاثير حيث مفاتيح النظام لا يجب اغلاقها استدعاء هذه الدالة من خلال نسخة من الكائن (instance) الذي يكون مغلق لا يؤثر شي

مثال :-

مثال	الغة
<pre>Console.WriteLine(vbCrLf & "Delete newly created " & _ "registry key? (Y/N) ") If Char.ToUpper(Convert.ToChar(Console.Read())) = "Y" Then</pre>	VB

<pre> Registry.CurrentUser.DeleteSubKeyTree("Test9999") Console.WriteLine(vbCrLf & "Registry key {0} deleted.", _ test9999.Name) Else Console.WriteLine(vbCrLf & "Registry key {0} closed.", _ test9999.ToString()) test9999.Close() End If </pre>	
<pre> Console.WriteLine("\nDelete newly created registry key? (Y/N) "); if(Char.ToUpper(Convert.ToChar(Console.Read())) == 'Y') { Registry.CurrentUser.DeleteSubKeyTree("Test9999"); Console.WriteLine("\nRegistry key {0} deleted.", test9999.Name); } else { Console.WriteLine("\nRegistry key {0} closed.", test9999.ToString()); test9999.Close(); } </pre>	C#
<pre> Console::Write(S"\nDelete newly created registry key? (Y/N) "); if(Char::ToUpper(Convert::ToChar(Console::Read())) == 'Y') { Registry::CurrentUser->DeleteSubKeyTree(S"Test9999"); Console::WriteLine(S"\nRegistry key {0} deleted.", test9999->Name); } else { Console::WriteLine(S"\nRegistry key {0} closed.", test9999->ToString()); test9999->Close(); } </pre>	C++

هذا الكود حسب **.NET Framework version 1.1.**

الدالة **RegistryKey.CreateSubKey** :

المهمة :- انشاء مفتاح جديد او فتح مفتاح موجود .

الصيغة

اللغة	الصيغة
VB.NET	<code>Public Function CreateSubKey(ByVal subkey As String, ByVal registryKey As RegistryKey) As RegistryKey</code>
C#	<code>public static RegistryKey CreateSubKey(RegistryKey registryKey, string subkey)</code>
C++	<code>public static RegistryKey CreateSubKey(RegistryKey registryKey, const string& subkey)</code>
JScript	<code>function CreateSubKey(registryKey, subkey)</code>

المتغيرات (Parameters) :-

SubKey :- اسم او مسار المفتاح المراد انشاءه او فتحة (اسم المفتاح غير حساس لحالة الاحرف).

القيمة المرجعة (RegistryKey) :- ترجع المفتاح الفرعي او قيمة غير معرفة (NULL) في حالة فشل الدالة

أخفاً متوقعة عند استخدام هذه الدالة :-

نوع الخطأ	حالة الوقوع
ArgumentNullException	عندما يكون المفتاح الفرعي غير معرف (NULL)
SecurityException	المستخدم لا يملك صلاحيات للتعديل على الرجستري
ArgumentException	طول المفتاح الفرعي اكبر من الحجم المحدد 255 حرف
ObjectDisposedException	المفتاح الرئيسي تم اغلاقه (لا يمكن الوصول الى المفتاح المغلق)
UnauthorizedAccessException	لا يمكن التعديل على المفتاح الرئيسي (مثال تم فتح المفتاح الرئيسي بشكل غير قابل للتعديل)

ملاحظة :-

لاستخدام هذه الدالة لابد ان يكون للمستخدم صلاحيات للوصول الى الرجستري و التعديل عليها .

مثال :-

اللغة	المثال
VB.NET	<pre> Public Class RegKey Shared Sub Main() Dim test9999 As RegistryKey = _ Registry.CurrentUser.CreateSubKey("Test9999") test9999.CreateSubKey("TestName").Close() Dim testSettings As RegistryKey = _ test9999.CreateSubKey("TestSettings") testSettings.SetValue("Language", "French") testSettings.SetValue("Level", "Intermediate") testSettings.SetValue("ID", 123) testSettings.SetValue("Password", "Secret") testSettings.Close() Console.WriteLine("There are {0} subkeys under Test9999.", _ test9999.SubKeyCount.ToString()) For Each subKeyName As String In test9999.GetSubKeyNames() Dim tempKey As RegistryKey = _ test9999.OpenSubKey(subKeyName) Console.WriteLine(vbCrLf & "There are {0} values for " & _ "{1}.", tempKey.ValueCount.ToString(), tempKey.Name) For Each valueName As String In tempKey.GetValueNames() Console.WriteLine("{0,-8}: {1}", valueName, _ tempKey.GetValue(valueName).ToString()) Next Next ' Delete the non-secure password value. testSettings = test9999.OpenSubKey("TestSettings", True) testSettings.DeleteValue("password") ' Verify the deletion. Console.WriteLine(CType(testSettings.GetValue(_ "password", "Password not found."), String)) testSettings.Close() ' Delete or close the new subkey. Console.WriteLine(vbCrLf & "Delete newly created " & _ "registry key? (Y/N) ") If Char.ToUpper(Convert.ToChar(Console.Read())) = "Y" Then Registry.CurrentUser.DeleteSubKeyTree("Test9999") Console.WriteLine(vbCrLf & "Registry key {0} deleted.", _ test9999.Name) End If End Sub End Class </pre>

<pre> Else Console.WriteLine(vbCrLf & "Registry key {0} closed.", _ test9999.ToString()) test9999.Close() End If End Sub End Class </pre>	
<pre> using System; using System.Security.Permissions; using Microsoft.Win32; [assembly: RegistryPermissionAttribute(SecurityAction.RequestMinimum, All = "HKEY_CURRENT_USER")] class RegKey { static void Main() { RegistryKey test9999 = Registry.CurrentUser.CreateSubKey("Test9999"); using(RegistryKey testName = test9999.CreateSubKey("TestName"), testSettings = test9999.CreateSubKey("TestSettings")) { testSettings.SetValue("Language", "French"); testSettings.SetValue("Level", "Intermediate"); testSettings.SetValue("ID", 123); testSettings.SetValue("Password", "Secret"); } Console.WriteLine("There are {0} subkeys under {1}.", test9999.SubKeyCount.ToString(), test9999.Name); foreach(string subKeyName in test9999.GetSubKeyNames()) { using(RegistryKey tempKey = test9999.OpenSubKey(subKeyName)) { Console.WriteLine("\nThere are {0} values for {1}.", tempKey.ValueCount.ToString(), tempKey.Name); foreach(string valueName in tempKey.GetValueNames()) { Console.WriteLine("{0,-8}: {1}", valueName, tempKey.GetValue(valueName).ToString()); } } } using(RegistryKey testSettings = test9999.OpenSubKey("TestSettings", true)) { testSettings.DeleteValue("password"); Console.WriteLine((string)testSettings.GetValue("password", "Password not found.)); } Console.Write("\nDelete newly created registry key? (Y/N) "); if(Char.ToUpper(Convert.ToChar(Console.Read())) == 'Y') { Registry.CurrentUser.DeleteSubKeyTree("Test9999"); Console.WriteLine("\nRegistry key {0} deleted.", test9999.Name); } else </pre>	C#

<pre> Console.WriteLine("\nRegistry key {0} closed.", test9999.ToString()); test9999.Close(); } } } </pre>	
--	--

RegistryKey.DeleteSubKey (الدالة St)

ring)

المهمة : حذف المفتاح الفرعي من التسجيل

الصيغة :-

الصيغة	اللغة
<code>StringOverloads Public Sub DeleteSubKey(ByVal subkey As subkey);string DeleteSubKey(voidpublic</code>	VB.NET
<code>* subkey);String DeleteSubKey(voidpublic:</code>	C#
<code>);Stringpublic function DeleteSubKey(subkey :</code>	C++
	JScript

Parameters المتغيرات

- SubKey المفتاح الفرعي المراد حذفه (غير حساس لحالة الأحرف)
الخطأ المتوقع:-

الخطأ	السبب
InvalidOperationException	المفتاح الفرعي يحتوي على مفتاح فرعي آخر
ArgumentException	المفتاح الفرعي المحدد غير موجود
ArgumentNullException	المفتاح الفرعي المحدد غير معرف (NULL)
SecurityException	المستخدم لا يملك صلاحيات للتعديل على التسجيل

ملاحظة :- هذه الدالة تقوم بفحص المفتاح الفرعي إذا كان يحتوي على مفاتيح فرعية أخرى تحتها .

مثال

مثال	اللغة
<pre> Console.WriteLine(vbCrLf & "Delete newly created " & _ "registry key? (Y/N) ") If Char.ToUpper(Convert.ToChar(Console.Read())) = "Y" Then Registry.CurrentUser.DeleteSub ("Test9999") Console.WriteLine(vbCrLf & "Registry key {0} deleted.", _ test9999.Name) Else Console.WriteLine(vbCrLf & "Registry key {0} closed.", _ test9999.ToString()) test9999.Close() End If </pre>	VB
<pre> Console.WriteLine("\nDelete newly created registry key? (Y/N) "); if(Char.ToUpper(Convert.ToChar(Console.Read())) == 'Y') { Registry.CurrentUser.DeleteSubKey ("Test9999"); Console.WriteLine("\nRegistry key {0} deleted.", test9999.Name); } else { Console.WriteLine("\nRegistry key {0} closed.", test9999.ToString()); test9999.Close(); } </pre>	C#
<pre> Console::Write(S"\nDelete newly created registry key? (Y/N) "); if(Char::ToUpper(Convert::ToChar(Console::Read())) == 'Y') { Registry::CurrentUser->DeleteSubKey (S"Test9999"); Console::WriteLine(S"\nRegistry key {0} deleted.", test9999->Name); } else { Console::WriteLine(S"\nRegistry key {0} closed.", test9999->ToString()); test9999->Close(); } </pre>	C++

الدالة RegistryKey.DeleteSubKey Method (String, Boolean)

تعتبر هذه الدالة استنساخ من الدالة السابقة (OverLoad) و تقوم بحذف المفتاح الفرعي المحدد

الصيغة :-

الصيغة	اللغة
<pre> , _StringOverloads Public Sub DeleteSubKey(ByVal subkey As _Boolean ByVal throwOnMissingSubKey As </pre>	VB.NET

)	
	<code>string DeleteSubKey(voidpublic throwOnMissingSubKeyboolsubkey,);</code>	C#
	<code>* subkey,String DeleteSubKey(voidpublic: throwOnMissingSubKeybool);</code>	C++
	<code>,Stringpublic function DeleteSubKey(subkey :);Boolean throwOnMissingSubKey :</code>	JScript

المتغيرات (Parameters)

SubKey المفتاح الفرعي المراد حذفه

throwOnMissingSubKey تحديد اذا نريد ان يعطينا خطأ في حالة عدم العثور على المفتاح الفرعي المحدد . اذا كانت القيمة لهذا المتغير False لا يعطينا خطأ اذا لم يجد المفتاح الفرعي اما اذا كانت القيمة True فيعطينا خطأ .

الخطا المتوقعة :-

الخطا	السبب
InvalidOperationException	المفتاح الفرعي يحتوي مفاتيح فرعية اخرى تحته
ArgumentException	عدم العثور على المفتاح الفرعي داخل الرجستري و كانت قيمة المتغير الثاني في الدالة True
ArgumentNullException	اذا كان المفتاح الفرعي غير معرف (Null)
SecurityException	المستخدم لا يملك صلاحيات للتعديل على الرجستري

الدالة **RegistryKey.DeleteSubKeyTree**

المهمة :- حذف المفتاح الفرعي مع المفاتيح الاخرى التي تدرج تحته .

الصيغة :-

الصيغة	الفة
<code>Public Sub DeleteSubKeyTree(_String ByVal subkey As)</code>	VB.NET
<code>DeleteSubKeyTree(voidpublic subkeystring);</code>	C#
<code>DeleteSubKeyTree(voidpublic: * subkey);String</code>	C++
<code>public function DeleteSubKeyTree();String subkey :</code>	Jcript

المتغيرات (Parameters)

SubKey : المفتاح الفرعي المراد حذفه مع المفاتيح الاخرى التي تدرج تحته

الاطلا المتوقعة :-

الخطا	السبب
ArgumentNullException	المفتاح الفرعي غير معرف
ArgumentException	محاولة حذف مفتاح رئيسي
SecurityException	المستخدم لا يملك صلاحيات للتعديل على الرجستري

ملاحظة :-

يجب ان يملك المستخدم صلاحيات الحذف من الرجستري
هذه الدالة تقوم بحذف المفتاح الفرعي مع كل المفاتيح الفرعية التي تندرج تحت هذا المفتاح الفرعي . لا يظهر
البرنامج أي تحذير لهذه العملية . اذا كنت تريد حذف مفتاح فرعي لا يحتوي على مفاتيح اخرى تحتة استخدم الدالة
.deleteSubKey

مثال :-

الغة	المثال
VB.NET	<pre> Console.Write(vbCrLf & "Delete newly created " & _ "registry key? (Y/N) ") If Char.ToUpper(Convert.ToChar(Console.Read())) = "Y" Then Registry.CurrentUser.DeleteSubKeyTree("Test9999") Console.WriteLine(vbCrLf & "Registry key {0} deleted.", _ test9999.Name) Else Console.WriteLine(vbCrLf & "Registry key {0} closed.", _ test9999.ToString()) test9999.Close() End If </pre>
C#	<pre> Console.Write("\nDelete newly created registry key? (Y/N) "); if(Char.ToUpper(Convert.ToChar(Console.Read())) == 'Y') { Registry.CurrentUser.DeleteSubKeyTree("Test9999"); Console.WriteLine("\nRegistry key {0} deleted.", test9999.Name); } else { Console.WriteLine("\nRegistry key {0} closed.", test9999.ToString()); test9999.Close(); } </pre>
C++	<pre> Console::Write(S"\nDelete newly created registry key? (Y/N) "); if(Char::ToUpper(Convert::ToChar(Console::Read())) == 'Y') { Registry::CurrentUser->DeleteSubKeyTree(S"Test9999"); Console::WriteLine(S"\nRegistry key {0} deleted.", test9999->Name); } </pre>

<pre> else { Console.WriteLine(S"\nRegistry key {0} closed.", test9999->ToString()); test9999->Close(); } </pre>	
--	--

الدالة RegistryKey.DeleteValue (String)

المهمة: -- حذف قيمة المفتاح المحدد

الصيغة: --

الصيغة	الغة
<pre> Overloads Public Sub DeleteValue(_ _String ByVal name As) </pre>	VB.NET
<pre> DeleteValue(voidpublic namestring); </pre>	C#
<pre> DeleteValue(voidpublic: * name);String </pre>	C++
<pre> public function DeleteValue();String name : </pre>	JScript

المتغيرات: --

Name : اسم القيمة المراد حذفها

الاحطاء المتوقعة

الخطا	السبب
ArgumentException	الاسم المحدد غير موجود او غير معرف
SecurityException	المستخدم لا يملك صلاحيات التعديل على الرجستر
ObjectDisposedException	المفتاح المستخدم للحذف مغلق

لا يمكن
التعديل
على
المفتاح
لأنه
للقراءة
فقط

UnauthorizedAccessException

مثال

المثال	اللغة
<pre>' Delete the non-secure password value. testSettings = test9999.OpenSubKey("TestSettings", True) testSettings.DeleteValue("password") ' Verify the deletion. Console.WriteLine(CType(testSettings.GetValue(_ "password", "Password not found."), String)) testSettings.Close()</pre>	VN.NET
<pre>using(RegistryKey testSettings = test9999.OpenSubKey("TestSettings", true)) { // Delete the non-secure password value. testSettings.DeleteValue("password"); // Verify the deletion. Console.WriteLine((string)testSettings.GetValue("password", "Password not found.)); }</pre>	C#
<pre>// Delete the non-secure password value. testSettings = test9999- >OpenSubKey(S"TestSettings", true); testSettings->DeleteValue(S"password"); // Verify the deletion. Console::WriteLine(dynamic_cast<String*>(testSettings->GetValue("password", S"Password not found.))); testSettings->Close();</pre>	C++

الدالة RegistryKey.DeleteValue (String, Boolean)

تعتبر هذه الدالة مستنسخة من الدالة السابقة ولها نفس المهمة

الصيغة

الصيغة	اللغة
<pre>Overloads Public Sub DeleteValue(_ ,_String ByVal name As _Boolean ByVal throwOnMissingValue As)</pre>	VB.NET
<pre>DeleteValue(voidpublic name,string throwOnMissingValuebool);</pre>	C#
<pre>DeleteValue(voidpublic: * name,String throwOnMissingValuebool);</pre>	C++
<pre>public function DeleteValue(,String name :);Boolean throwOnMissingValue :</pre>	JScript

المتغيرات

Name :- اسم القيمة المراد حذفها من المفتاح
throwOnMissingValue :- لتحديد اذا اردنا اظهار خطأ في حالة عدم العثور على اسم القيمة . اذا كانت القيمة True يظهر الخطأ اذا لم يجد الاسم . اذا كانت القيمة false لا يظهر الخطأ

الايخطاء المتوقعة :-

الخطا	السبب
ArgumentException	عدم العثور على القيمة و المتغير الثاني كان True او القيمة المحدد للاسم غير معرفة
SecurityException	المستخدم لا يملك صلاحيات التعديل على الرجستري
ObjectDisposedException	محولة الحذف عن طريق مفتاح مغلق
UnauthorizedAccessException	محولة الحذف على مفتاح للقراءة فقط

ملاحظة :-

لا يوجد طريقة للتأكد من ان البرنامج وجد القيمة و قام بحذفها في حالة ان المتغير الثاني كان false لذلك يفضل التأكد من ذلك عن طريق Try catch

الدالة RegistryKey.GetSubKeyNames

المهمة :- ترجع مصفوفة تحتوي على اسماء كل المفاتيح الفرعية

الصيغة :-

الصيغة	اللغة
<pre>StringPublic Function GetSubKeyNames() As ()</pre>	VB.NET
<pre>[] GetSubKeyNames();stringpublic</pre>	C#

* <code>GetSubKeyNames()</code> <code>__gc[];String</code> public:	C++
<code>[];String</code> public function <code>GetSubKeyNames()</code> :	JScript

القيمة المرجعة :- مصفوفة من المتسلسلة الحرفية التي تمثل اسماء كل المفاتيح الفرعية

الايخطاء المتوقعة :

الخطا	السبب
SecurityException	المستخدم لا يملك صلاحيات القراءة من الرجستري
ObjectDisposedException	القراءة من مفتاح مغلق

هذه الدالة لا تستدعي ذاتيا لذلك هي تعرض اسماء المفاتيح الفرعية في مستوى واحد فقط

مثال :

المثال	الغة
<pre>' Print the information from the Test9999 subkey. Console.WriteLine("There are {0} subkeys under Test9999.", _ test9999.SubKeyCount.ToString()) For Each subKeyName As String In test9999.GetSubKeyNames() Dim tempKey As RegistryKey = _ test9999.OpenSubKey(subKeyName) Console.WriteLine(vbCrLf & "There are {0} values for " & _ "{1}.", tempKey.ValueCount.ToString(), tempKey.Name) For Each valueName As String In tempKey.GetValueNames() Console.WriteLine("{0,-8}: {1}", valueName, _ tempKey.GetValue(valueName).ToString()) Next Next</pre>	VB.NET
<pre>// Print the information from the Test9999 subkey. Console.WriteLine("There are {0} subkeys under {1}.", test9999.SubKeyCount.ToString(), test9999.Name); foreach(string subKeyName in test9999.GetSubKeyNames()) { using(RegistryKey tempKey = test9999.OpenSubKey(subKeyName)) { Console.WriteLine("\nThere are {0} values for {1}.", tempKey.ValueCount.ToString(), tempKey.Name); foreach(string valueName in tempKey.GetValueNames()) { Console.WriteLine("{0,-8}: {1}", valueName, tempKey.GetValue(valueName).ToString()); } } }</pre>	C#

<pre> } } } </pre>	
<pre> // Print the information from the Test9999 subkey. Console::WriteLine(S"There are {0} subkeys under Test9999.", test9999->SubKeyCount.ToString()); String* subKeyNames __gc [] = test9999->GetSubKeyNames(); for(int i =0; i < subKeyNames->Length; i++) { RegistryKey* tempKey = test9999- >OpenSubKey(subKeyNames[i]); Console::WriteLine(S"\nThere are {0} values for {1}.", tempKey->ValueCount.ToString(), tempKey->Name); String* valueNames __gc [] = tempKey- >GetValueNames(); for(int j = 0; j < valueNames->Length; j++) { Console::WriteLine(S"{0,-8}: {1}", valueNames[j], tempKey->GetValue(valueNames[j])- >ToString()); } } </pre>	C++

الدالة RegistryKey.OpenSubKey (String)

المهمة :- ارجاع مفتاح فرعي للقراءة فقط

الصيغة :-

الصيغة	اللغة
<pre> Overloads Public Function OpenSubKey(_ _String ByVal name As RegistryKey) As </pre>	VB.Net
<pre> OpenSubKey(RegistryKeypublic namestring); </pre>	C#
<pre> * OpenSubKey(RegistryKeypublic: * nameString); </pre>	C++

<pre> public function OpenSubKey(String name : ;RegistryKey) :</pre>	JScript

المتغيرات

Name اسم او مسار المفتاح المراد فتحه
القيمة المرجعة :- المفتاح الذي تم فتحه او قيمة غير معرفة في حالة فشل الدالة

الايخطاء المتوقعة

الخطا	السبب
ArgumentNullException	الاسم غير معرف
ArgumentException	اسم المفتاح اكبر من الحجم المحدد 255 حرف
ObjectDisposedException	المفتاح الرئيسي مغلق
SecurityException	المستخدم لا يملك صلاحيات القراءة من الجستري

ملاحظات

يجب فتح المفتاح اولا قبل عملية التعديل عليه
اذا لم يتم العثور على المفتاح المحدد سوف تكون القيمة المرجعة غير معرفة

مثال

المثال

```

' Print the information from the
Test9999 subkey.
Console.WriteLine("There are {0} subkeys
under Test9999.", _
test9999.SubKeyCount.ToString())
For Each subKeyName As String In
test9999.GetSubKeyNames()
Dim tempKey As RegistryKey = _
test9999.OpenSubKey(subKeyName)
Console.WriteLine(vbCrLf & "There
are {0} values for " & _
"{1}.",
tempKey.ValueCount.ToString(),
tempKey.Name)
For Each valueName As String In
tempKey.GetValueNames()
Console.WriteLine("{0,-8}: {1}",
valueName, _
```

الغة
VB.NET

```

tempKey.GetValue(valueName).ToString())
        Next
    Next

    // Print the information from the
    Test9999 subkey.
Console.WriteLine("There are {0} subkeys
    under {1}.",
    test9999.SubKeyCount.ToString(),
    test9999.Name);
    foreach(string subKeyName in
        test9999.GetSubKeyNames())
    {
        using(RegistryKey
            tempKey =
test9999.OpenSubKey(subKeyName))
        {
            Console.WriteLine("\nThere are
                {0} values for {1}.",

                tempKey.ValueCount.ToString(),
                tempKey.Name);
            foreach(string valueName in
                tempKey.GetValueNames())
            {
                Console.WriteLine("{0,-8}:
                    {1}", valueName,

tempKey.GetValue(valueName).ToString());
            }
        }
    }
}

```

C#

```

// Print the information from the
    Test9999 subkey.
Console::WriteLine(S"There are {0}
    subkeys under Test9999.",
    test9999->SubKeyCount.ToString());
String* subKeyNames __gc [] = test9999-
    >GetSubKeyNames();
for(int i =0; i < subKeyNames->Length;
    i++)
{
    RegistryKey* tempKey = test9999-
        >OpenSubKey(subKeyNames[i]);
    Console::WriteLine(S"\nThere are {0}
        values for {1}.",
        tempKey->ValueCount.ToString(),
        tempKey->Name);
    String* valueNames __gc [] =
        tempKey->GetValueNames();
    for(int j = 0; j < valueNames-
        >Length; j++)
    {
        Console::WriteLine(S"{0,-8}:
            {1}", valueNames[j],

```

C++

```

tempKey-
>GetValue(valueNames[j])->ToString();
}
}

```

الدالة RegistryKey.OpenSubKey (String, Boolean)

المهمة ارجاع المفتاح المحدد

الصيغة:-

الصيغة	اللغة
<pre> Overloads Public Function OpenSubKey(_ ,_String ByVal name As _Boolean ByVal writable As RegistryKey) As </pre>	VB.NET
<pre> OpenSubKey(RegistryKeypublic name,string writablebool); </pre>	C#
<pre> * OpenSubKey(RegistryKeypublic: * name,String writablebool); </pre>	C++
<pre> public function OpenSubKey(,String name : Boolean writable : ;RegistryKey): </pre>	JScript

المتغيرات

Name :- اسم المفتاح المراد ارجاعه
Writable :- قابل للتعديل ام لا اذا كانت القيمة True يكون قابل للتعديل ام اذا كانت القيمة false يكون غير قابل للتعديل

القيمة المرجعة
المفتاح المراد فتحة

اخطاء متوقعة

الخطا	السبب
ArgumentNullException	الاسم غير معرف
ArgumentException	الاسم المحدد اكبر من الحجم المسموح 255 حرف

المفتاح الرئيسي مغلق	ObjectDisposedException
ليس للمستخدم صلاحية القراءة من الجسري	SecurityException

ملاحظات

يقوم البرنامج بإصدار خطأ في حالة عدم العثور على المفتاح المحدد بالإضافة لذلك يرجع قيمة غير معرفة

إذا كان قيمة المتغير الثاني True يمكن قراءة المفتاح و التعديل عليه غير ذلك يكون للقراءة فقط

مثال

الغة	مثال
VB.NET	<pre> ' Delete the non-secure password value. testSettings = test9999.OpenSubKey("TestSettings", True) testSettings.DeleteValue("password") ' Verify the deletion. Console.WriteLine(CType(testSettings.GetValue(_ "password", "Password not found."), String)) testSettings.Close()</pre>
C#	<pre> using(RegistryKey testSettings = test9999.OpenSubKey("TestSettings", true)) { // Delete the non-secure password value. testSettings.DeleteValue("password"); // Verify the deletion. Console.WriteLine((string)testSettings.GetValue("password", "Password not found.)); }</pre>
C++	<pre> using(RegistryKey testSettings = test9999.OpenSubKey("TestSettings", true)) { // Delete the non-secure password value. testSettings.DeleteValue("password"); // Verify the deletion. Console.WriteLine((string)testSettings.GetValue("password", "Password not found.)); }</pre>

الدالة RegistryKey.SetValue

المهمة :- التعديل على قيمة معينة

الصيغة:-

الغة	الصيغة
VB.NET	<pre>Public Sub SetValue(_ , _String ByVal name As _Object ByVal value As)</pre>
C#	<pre>SetValue(voidpublic name,string valueobject);</pre>
C++	<pre>SetValue(voidpublic: * name,String * valueObject);</pre>
JScript	<pre>public function SetValue(,String name :);Object value :</pre>

المتغيرات :-

Name :- اسم القيمة المراد تعديلها
Value :- القيمة الجديد

اخطا متوقعة

المستخدم لا يملك صلاحية التعديل على الراجستري	SecurityException
القيمة الجديد غير معروفة	ArgumentNullException
القيمة الجديد اكبر من الحجم المسموح به	ArgumentException
المفتاح الفرعي مغلق	ObjectDisposedException
المفتاح الفرعي للقراءة فقط	UnauthorizedAccessException

ملاحظات:-

لان المفتاح بمك من الاحتفاظ باكثر من قيمة الاسم يحدد أي من هذه القيم سوف يتم عليها التعديل لارجاع القيمة الافتراضية لقيمة معينة يمكن وضع قيمة غير معرفة غي المتغير الاول (Name) او سلسلة حرفية فارغة

القيمة داخل المفتاح يمكن ان تكون ثنائية, dWord, او سلسلة حرفية

المفتاح المراد تعديله يجب ان يكون مفتوح للكتابة و القراءة

اذا لم يتم العثور على المفتاح يقوم بنشاء قيمة جديد مع اعطائها القيمة المحددة

مثال

مثال	الغة
<pre>Imports Microsoft.VisualBasic Imports System Imports System.Security.Permissions Imports Microsoft.Win32 <Assembly: RegistryPermissionAttribute(_ SecurityAction.RequestMinimum, All := "HKEY_CURRENT_USER")> Public Class RegKey Shared Sub Main() ' Create a subkey named Test9999 under HKEY_CURRENT_USER. Dim test9999 As RegistryKey = _ Registry.CurrentUser.CreateSubKey("Test9999") ' Create two subkeys under HKEY_CURRENT_USER\Test9999. test9999.CreateSubKey("TestName").Close() Dim testSettings As RegistryKey = _ test9999.CreateSubKey("TestSettings") ' Create data for the TestSettings subkey. testSettings.SetValue("Language", "French") testSettings.SetValue("Level", "Intermediate") testSettings.SetValue("ID", 123) testSettings.SetValue("Password", "Secret") testSettings.Close() ' Print the information from the Test9999 subkey. Console.WriteLine("There are {0} subkeys under Test9999.", _ test9999.SubKeyCount.ToString()) For Each subKeyName As String In test9999.GetSubKeyNames() Dim tempKey As RegistryKey = _ test9999.OpenSubKey(subKeyName) Console.WriteLine(vbCrLf & "There are {0} values for " & _ "{1}.", tempKey.ValueCount.ToString(), tempKey.Name) For Each valueName As String In</pre>	VB.NET

<pre> tempKey.GetValueNames() Console.WriteLine("{0,-8}: {1}", valueName, tempKey.GetValue(valueName).ToString()) Next Next ' Delete the non-secure password value. testSettings = test9999.OpenSubKey("TestSettings", True) testSettings.DeleteValue("password") ' Verify the deletion. Console.WriteLine(CType(testSettings.GetValue("password", "Password not found."), String)) testSettings.Close() ' Delete or close the new subkey. Console.Write(vbCrLf & "Delete newly created " & _ "registry key? (Y/N) ") If Char.ToUpper(Convert.ToChar(Console.Read())) = "Y" Then Registry.CurrentUser.DeleteSubKeyTree("Test9999") Console.WriteLine(vbCrLf & "Registry key {0} deleted.", _ test9999.Name) Else Console.WriteLine(vbCrLf & "Registry key {0} closed.", _ test9999.ToString()) test9999.Close() End If End Sub End Class </pre>	
<pre> using System; using System.Security.Permissions; using Microsoft.Win32; [assembly: RegistryPermissionAttribute(SecurityAction.RequestMinimum, All = "HKEY_CURRENT_USER")] class RegKey { static void Main() { // Create a subkey named Test9999 under HKEY_CURRENT_USER. RegistryKey test9999 = Registry.CurrentUser.CreateSubKey("Test9999"); // Create two subkeys under HKEY_CURRENT_USER\Test9999. The // keys are disposed when execution exits the using statement. </pre>	C#

```

        using(RegistryKey
testName = test9999.CreateSubKey("TestName"),
        testSettings =
test9999.CreateSubKey("TestSettings"))
    {
        // Create data for the TestSettings subkey.
testSettings.SetValue("Language", "French");
testSettings.SetValue("Level", "Intermediate");
testSettings.SetValue("ID", 123);
testSettings.SetValue("Password", "Secret");
    }

// Print the information from the Test9999 subkey.
Console.WriteLine("There are {0} subkeys under
    {1}.",
test9999.SubKeyCount.ToString(),
test9999.Name);
foreach(string subKeyName in
test9999.GetSubKeyNames())
{
    using(RegistryKey
tempKey = test9999.OpenSubKey(subKeyName))
    {
        Console.WriteLine("\nThere are {0} values
            for {1}.",
tempKey.ValueCount.ToString(),
tempKey.Name);
        foreach(string valueName in
tempKey.GetValueNames())
        {
            Console.WriteLine("{0,-8}: {1}",
valueName,

tempKey.GetValue(valueName).ToString());
        }
    }

    using(RegistryKey
testSettings =
test9999.OpenSubKey("TestSettings", true))
    {
        // Delete the non-secure password value.
testSettings.DeleteValue("password");

        // Verify the deletion.

Console.WriteLine((string)testSettings.GetValue(
    "password", "Password not found.));

        // Delete or close the new subkey.
Console.WriteLine("\nDelete newly created registry key?
    (Y/N) ");
if(Char.ToUpper(Convert.ToChar(Console.Read())) ==
    'Y')
    {

```


<pre> Registry.CurrentUser.DeleteSubKeyTree("Test9999"); Console.WriteLine("\nRegistry key {0} deleted.", test9999.Name); } else { Console.WriteLine("\nRegistry key {0} closed.", test9999.ToString()); test9999.Close(); } } } } </pre>	
<pre> #using <mscorlib.dll> using namespace System; using namespace System::Security::Permissions; using namespace Microsoft::Win32; [assembly: RegistryPermissionAttribute(SecurityAction::RequestMinimum, All = S"HKEY_CURRENT_USER")]; void main() { // Create a subkey named Test9999 under HKEY_CURRENT_USER. RegistryKey* test9999 = Registry::CurrentUser->CreateSubKey(S"Test9999"); // Create two subkeys under HKEY_CURRENT_USER\Test9999. test9999->CreateSubKey(S"TestName")->Close(); RegistryKey* testSettings = test9999->CreateSubKey(S"TestSettings"); // Create data for the TestSettings subkey. testSettings->SetValue(S"Language", S"French"); testSettings->SetValue(S"Level", S"Intermediate"); testSettings->SetValue(S"ID", __box(123)); testSettings->SetValue(S"Password", S"Secret"); testSettings->Close(); // Print the information from the Test9999 subkey. Console::WriteLine(S"There are {0} subkeys under Test9999.", test9999->SubKeyCount.ToString()); String* subKeyNames __gc [] = test9999- >GetSubKeyNames(); for(int i =0; i < subKeyNames->Length; i++) { RegistryKey* tempKey = test9999- >OpenSubKey(subKeyNames[i]); Console::WriteLine(S"\nThere are {0} values for {1}.", tempKey->ValueCount.ToString(), tempKey->Name); } } </pre>	C++

```

        String* valueNames __gc [] = tempKey-
            >GetValueNames();
        for(int j = 0; j < valueNames->Length; j++)
        {
            Console::WriteLine(S"{0,-8}: {1}",
                valueNames[j],
                tempKey->GetValue(valueNames[j])-
                    >ToString());
        }
    }

    // Delete the non-secure password value.
    testSettings = test9999->OpenSubKey(S"TestSettings",
        true);
    testSettings->DeleteValue(S"password");

    // Verify the deletion.
    Console::WriteLine(dynamic_cast<String*>(
        testSettings->GetValue("password", S"Password not
            found.")));
    testSettings->Close();

    // Delete or close the new subkey.
    Console::Write(S"\nDelete newly created registry key?
        (Y/N) ");
    if(Char::ToUpper(Convert::ToChar(Console::Read())) ==
        'Y')
    {
        Registry::CurrentUser-
            >DeleteSubKeyTree(S"Test9999");
        Console::WriteLine(S"\nRegistry key {0} deleted.",
            test9999->Name);
    }
    else
    {
        Console::WriteLine(S"\nRegistry key {0} closed.",
            test9999->ToString());
        test9999->Close();
    }
}

```

الدالة RegistryKey.OpenRemoteBaseKey

المهمة :- فتح المفتاح المحدد في الرجستري للجهاز المحدد

الصيغة :-

الغة	الصيغة
VB.NET	<pre>Public Shared Function OpenRemoteBaseKey(_ , _RegistryHive ByVal hKey As RegistryKey) As String ByVal machineName As</pre>
C#	<pre>OpenRemoteBaseKey(RegistryKeypublic static hKey,RegistryHive machineNamestring);</pre>
C++	<pre>* OpenRemoteBaseKey(RegistryKeypublic: static hKey,RegistryHive * machineNameString);</pre>
JScript	<pre>public static function OpenRemoteBaseKey(,RegistryHive hKey : ;RegistryKey) : String machineName :</pre>

المتغيرات :-

Hkey :- اسم المفتاح المراد فتحة

machineName :- اسم الجهاز

hKey :- اسم المفتاح

machineName :- اسم الجهاز

hKey :- اسم المفتاح

الخطا	السبب
ArgumentException	اسم الجهاز خطأ او اسم المفتاح خطأ
ArgumentNullException	اسم الجهاز غير معرف
SecurityException	المستخدم ليس لديه صلاحيات

إذا لم يتم بعث اسم الجهاز سوف يقوم بفتح الرجستري على نفس الجهاز

اسم المفتاح يجب ان يكون مفتاح جذر حسب الصيغة لل **RegistryHive**

في حالة استخدام هذه الدالة كلا الجهازين يجب ان يحتوا على **remote registry service** و يملك **remote administration**

مثال

المثال	اللغة
<pre>Imports Microsoft.VisualBasic Imports System Imports System.IO Imports System.Security.Permissions Imports Microsoft.Win32 <Assembly: RegistryPermissionAttribute(_ SecurityAction.RequestMinimum, _ Read := "HKEY_CURRENT_USER\Environment")> <Assembly: SecurityPermissionAttribute(_ SecurityAction.RequestMinimum, UnmanagedCode := True)> Public Class RemoteKey Shared Sub Main(commandLineArgs As String()) Dim environmentKey As RegistryKey ' Check that an argument was specified when the ' program was invoked. If commandLineArgs.Length = 0 Then Console.WriteLine("Error: The name of the remote " & _ "computer must be specified as input on the " & _ "command line.") Return End If Try ' Open HKEY_CURRENT_USER\Environment on a ' remote computer. environmentKey = RegistryKey.OpenRemoteBaseKey(RegistryHive.CurrentUser, _ commandLineArgs(0)).OpenSubKey("Environment") Catch ex As IOException Console.WriteLine("{0}: {1}", _ ex.GetType().Name, ex.Message) Return End Try ' Print the values. Console.WriteLine("\nThere are {0} values For {1}.", _ environmentKey.ValueCount.ToString(), environmentKey.Name)</pre>	VB.NET

<pre> For Each valueName As String In environmentKey.GetValueNames() Console.WriteLine("{0,-20}: {1}", valueName, _ environmentKey.GetValue(valueName).ToString()) Next ' Close the registry key. environmentKey.Close() End Sub End Class </pre>	
<pre> using System; using System.IO; using System.Security.Permissions; using Microsoft.Win32; [assembly: RegistryPermissionAttribute(SecurityAction.RequestMinimum, Read = @"HKEY_CURRENT_USER\Environment")] [assembly: SecurityPermissionAttribute(SecurityAction.RequestMinimum, UnmanagedCode = true)] class RemoteKey { static void Main(string[] args) { RegistryKey environmentKey; string remoteName; // Check that an argument was specified when the // program was invoked. if(args.Length == 0) { Console.WriteLine("Error: The name of the remote " + "computer must be specified when the program is " + "invoked."); return; } else { remoteName = args[0]; } try { // Open HKEY_CURRENT_USER\Environment // on a remote computer. environmentKey = RegistryKey.OpenRemoteBaseKey(RegistryHive.CurrentUser, remoteName).OpenSubKey(</pre>	C#

<pre> "Environment"); } catch(IOException e) { Console.WriteLine("{0}: {1}", e.GetType().Name, e.Message); return; } // Print the values. Console.WriteLine("\nThere are {0} values for {1}.", environmentKey.ValueCount.ToString(), environmentKey.Name); foreach(string valueName in environmentKey.GetValueNames()) { Console.WriteLine("{0,-20}: {1}", valueName, environmentKey.GetValue(valueName).ToString()); } // Close the registry key. environmentKey.Close(); } } </pre>	
<pre> #using <mscorlib.dll> using namespace System; using namespace System::IO; using namespace System::Security::Permissions; using namespace Microsoft::Win32; [assembly: RegistryPermissionAttribute(SecurityAction::RequestMinimum, Read = "HKEY_CURRENT_USER\\Environment")]; [assembly: SecurityPermissionAttribute(SecurityAction::RequestMinimum, UnmanagedCode = true)]; void main(int argc, char* argv[]) { RegistryKey* environmentKey; // Check that an argument was specified when the // program was invoked. if(argc == 1) { Console::WriteLine(S"Error: The name of the remote computer " S"must be specified as input on the command line."); return; } } } </pre>	C++

```

try
{
    // Open HKEY_CURRENT_USER\Environment on a remote
    computer.
    environmentKey = RegistryKey::OpenRemoteBaseKey(
        RegistryHive::CurrentUser,
        argv[1])->OpenSubKey(S"Environment");
}
catch(IOException* e)
{
    Console::WriteLine("{0}: {1}",
        e->GetType()->Name, e->Message);
    return;
}

// Print the values.
Console::WriteLine(S"\nThere are {0} values for {1}.",
    environmentKey->ValueCount.ToString(),
    environmentKey->Name);
String* valueNames __gc [] = environmentKey-
    >GetValueNames();
for(int i = 0; i < environmentKey->ValueCount; i++)
{
    Console::WriteLine("{0,-20}: {1}", valueNames[i],
        environmentKey->GetValue(valueNames[i])-
            >ToString());
}

// Close the registry key.
environmentKey->Close();
}

```

اتمنى ان اكون قد اوصلت ولو القليل الى كل مسلم امن بالله ورسوله(صلى الله عليه وسلم)
 الان انا مشغول شوي في العمل والدراسه انشاء الله سوف اكمل ما بدأت به ارجو من
 الله عز وجل التقدم والنجاح.
 لا تنسوننا من الدعاء لي وللوالدين ولجميع امه سيدنا محمد صلى الله عليه وسلم.